Question4

Solution :

1. Setup

Let us denote $dp(d, l)$ is the maximum total weight end at node d with exactly length l, $w(s, d)$ is the weight of the edge from $s\ to\ d$, $DP(j)$ is the maximum total weight with length equal to l.

2. Subproblems

To find the path with maximum total weight with length l, we can denote there is an end node d, so the maximum total weight is:

$$DP(l) = max\{dp(d, l)\}, \qquad for\ all\ d\ \in G$$

And we can compute each $dp(d, l)$ by compute $max\{dp(d', l - 1) + w(d', d)\}$ for each d' has an edge to d.

Hence, the subproblems are 'the path with maximum total weight with length l' end at node d'', it's $dp(d', l - 1)$; $for\ d' \in G\ and\ l' < l$.

3. Build-up order

Solve the subproblems in the order

$$dp(d, 0), dp(d, 1), dp(d, 2), \ldots dp(d, l); \ \ for\ all\ d\ \in G$$

4. base case

$$dp(d, 0) = 0; for\ all\ d\ \in G$$

5. Recursion

Assume we had solved all the subproblem that we got all $dp(g, t); for\ all\ s, d\ \in G; t < l'$.

Then we need to compute $dp(d, l')$, we can compute by each node d' which has a path from d' to d:

$dp(d, l') =$

$$\max\{dp(d', l' - 1) + w(d', d)\}; if\ Edge(d', d)\ exist; for\ all\ d, d' \ \in G$$

6. Final solution

$$DP(K) = max\{dp(d, K)\},\ for\ all\ d\ \in G.$$

7. Time complexity

There are K subproblems and each subproblems is $O(VE)$, hence the overall time complexity of the algorithm is $O(KVE)$.